UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/538,764 | 06/10/2005 | Kenji Sato | WIP011 | 5565 |

25271 7590 02/07/2008

GALLAGHER & LATHROP, A PROFESSIONAL CORPORATION
601 CALIFORNIA ST
SUITE 1111
SAN FRANCISCO, CA 94108

| EXAMINER | |
|---|---|
| KHATRI, ANIL | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2191 | |

| MAIL DATE | DELIVERY MODE |
|---|---|
| 02/07/2008 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

| Office Action Summary | Application No. | Applicant(s) | |
|---|---|---|---|
| | 10/538,764 | SATO, KENJI | |
| | Examiner | Art Unit | |
| | Anil Khatri | 2191 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *10 June 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-36* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-36* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *10 June 2005* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *6/10/05*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

## DETAILED ACTION

### *Specification*

The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed.

The following title is suggested: "*Software Execution Control System for Distributing and Updating Software in Distributed Environment*".

### *Claim Rejections - 35 USC § 101*

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 1-36 are rejected under 35 USC 101 because they disclose a claimed invention that is an abstract idea as defined in the case *In re Warmerdam*, 33, F 3d 1354, 31 USPQ 2d 1754 (Fed. Cir. 1994).

*Analysis*: Claims 1-36 disclosed by the applicant as being a "software execution control system...". Since the claims are each a series of steps to be performed on a computer the processes must be analyzed to determine whether they are statutory under 35 USC 101.

Examiner interprets that the claims 1-36 are non-statutory because they do not disclose that how a system will be able to carry out its intended result. Applicant submit no substance that how this will be processed without incorporating a processor, memory and medium. Therefore, claims 1-36 are an abstract idea and merely manipulation of instructions for distribution and verification without further reciting steps that how it will be decoded and encoding therefore its

functionality cannot be realized. Thus, claims 1-36 are non-statutory and rejected under 35 USC 101.

Further, examiner interprets that claims 1-36 are non-statutory because claims recites computer program which are program, per se i.e. the description or expressions of the program are not physical things nor are they statutory process as they do not act being performed. Computer programs do not define any structural and functional interrelationship between the computer program and other claimed aspect of the invention which permits the computer program's functionality could be realized. Therefore, computer program is merely a set of instructions capable of being executed by a computer, the computer program itself is not a process.

***Claim Rejections - 35 USC § 112***

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claims 1-36 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

The claims are generally narrative and indefinite, failing to conform with current U.S. practice. They appear to be a literal translation into English from a foreign document and are replete with grammatical and idiomatic errors.

Claims 1-36 are rejected under 35 U.S.C. 112, second paragraph, as being incomplete for omitting essential elements, such omission amounting to a gap between the elements. See MPEP

§ 2172.01. The omitted elements are: processing, storing, how execution is done, decoding and encoding is done, start second software how, disable etc? Further, the language is unclear and vague.

***Claim Rejections - 35 USC § 102***

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1, 15, 21-24 and 31-32 are rejected under 35 U.S.C. 102(b) as being anticipated by *Chang et al* USPN 5,724,425.

Regarding claims 1, 15, 21-24 and 31-32

*Chang et al teaches,*

a distribution part which distributes second software that is encoded and an execution control program for controlling the execution of second software to user computer via a communications network (column 3, lines 15-35, This invention provides a method and apparatus utilizing public key encryption techniques for enhancing software security and for distributing software. The present invention includes a first computer which is provided with source code to be protected using the teachings of the present invention. In addition, a software application writer's private key, along with an application writer's license provided to the first computer. An application writer generally means a software company such as Microsoft Corporation, Adobe or Apple Computer, Inc. The application writer's license includes identifying information such as the application writer's name as well as the application

writer's public key. A compiler program executed by the first computer compiles the source code into binary code, and computes a message digest for the binary code. The first computer then encrypts the message digest using the application writer's private key, such that the encrypted message digest is defined as a digital "signature" of the application writer. A software passport is then generated which includes the application writer's digital signature, the application writer's license and the binary code. The software passport is then distributed to a user using any number of software distribution models known in the industry); and

a verification part which performs user verification by a request from execution control program installed in user computer, and which transmits specified information that is required in order to decode and start second software to execution control program via communications network in cases where it is confirmed that the user is a valid user (column 9, lines 19-25, The software produced by a licensed application writer will include a valid passport 50 (see FIGS. 5 and 6a) which contains a genuine writer's digital signature, and a valid application writer's license 52 issued by the platform builder. Any application writer who is not authorized by the platform builder will not possess a valid license. Therefore, the software passport generated by an unauthorized person will either have no valid license or no valid signature);

wherein second software is constructed so that this software can be started only by the starting information that is transferred from execution control program; and execution control program is

constructed so as to :decode encoded second software on the basis of specified information received from verification part, and substitute this second software for first software;

start second software by creating starting information on the basis of specified information and disable second software when the execution of second software is completed.(column 3, lines 38-65, A user, upon receipt of the software passport, loads the passport into a computer which determines whether the software passport includes the application writer's license and digital signature. In the event that the software passport does not include the application writer's license, or the application writer's digital signature, then the user's computer system discards the software passport and does not execute the binary code. As an additional security step, the user's computer computes a second message digest for the software passport and compares it to the first message digest, such that if the first and second message digests are not equal, the software passport is also rejected by the user's computer and the code is not executed.

If the first and second message digests are equal, the user's computer extracts the application writer's public key from the application writer's license for verification. The application writer's digital signature is decrypted using the application writer's public key. The user's computer then compares a message digest of the binary code to be executed, with the decrypted application writer's digital signature, such that if they are equal, the user's computer executes the binary code. Accordingly, software products distributed with the present invention's software passport permits the user's computer to authenticate the software as created by an authorized application writer who has been issued a valid application writer's license. Any unauthorized changes to the binary code comprising the distributed software is evident through the comparison of the calculated and encrypted message digests).

***Claim Rejections - 35 USC § 103***

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 2-14, 16-20, 25-30 and 33-36 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Chang et al* USPN 5,724,425 in view of *O'Neill* USPN 6,832,373

Regarding claims 2, 6-13 and 16-20

*Chang et al teaches,*

Execution control program is constructed but does not teach explicitly so that this program can handle a plurality of different types of second software, and specified information that is transmitted to execution control program by verification part includes storage destination address information and a starting argument for the second software to be started, and decoding key information for decoding the second software. However, *O'Neill teaches,* (column 17,lines 33-43, In the state 306, the update generator 102 pre-processes the existing image by searching for digital information sequences that will be used to build a hash table. The hash table includes a plurality of hash values that comprise addresses of particular digital sequences in the first code version that are stored in a data structure for subsequent lookup and retrieval. The hash values correspond to digital information sequences in the existing image, which may be used to build the newer code version. In one aspect, the hash array is formed from the existing code

version and identifies strings of digital information sequences in the existing code version. Further details of the hash array will be discussed in connection with FIG. 4 below). Therefore, it would have been obvious to a person of ordinary skill in the art at the time of the invention was made to incorporate verification process for address. The modification would have been obvious because one of ordinary skill in the art would have been motivated to combine teaching into software distribution and installed at correct destination to avoid errors.

Regarding claims 3-5

*Chang et al teaches*

execution control program transmits verification information including machine information that is specific to user computer and encrypting key information to verification part, verification part performs user verification on the basis of at least machine information, verification part encrypts specified information by means of encrypting key information and transmits this information to execution control program via said communications network in cases where it is confirmed that user is a valid user, and a plurality of sets of machine information can be registered in verification part (column 3,lines 15-36, This invention provides a method and apparatus utilizing public key encryption techniques for enhancing software security and for distributing software. The present invention includes a first computer which is provided with source code to be protected using the teachings of the present invention. In addition, a software application writer's private key, along with an application writer's license provided to the first computer. An application writer generally means a software company such as Microsoft Corporation, Adobe or Apple Computer, Inc. The application writer's license includes identifying information such as the application writer's name as well as the application

writer's public key. A compiler program executed by the first computer compiles the source code into binary code, and computes a message digest for the binary code. The first computer then encrypts the message digest using the application writer's private key, such that the encrypted message digest is defined as a digital "signature" of the application writer. A software passport is then generated which includes the application writer's digital signature, the application writer's license and the binary code. The software passport is then distributed to a user using any number of software distribution models known in the industry).

Regarding claims 14, 25, 30 and 33-36

*O'Neill teaches*

distribution part distributes a monitoring program to user computer together with second software and execution control program, said monitoring program respectively monitors the operating conditions of second software and execution control program, and respectively shuts down second software and execution control program, and also shuts itself down, in cases where either second software or execution control program stops operating, and execution control program respectively monitors the operating conditions of second software and monitoring program, and respectively shuts down second software and monitoring program, and also shuts itself down, in cases where either second software or monitoring program stops operating (column 13,lines 1-24, FIG. 2A illustrates an overview of an update query, retrieval and installation process or update installation process 200 that details the communication between the client devices and the update distribution system 140. The update installation process 200

commences in a start state 202 and subsequently proceeds to a state 204, where the client device 104 establishes a communication link with the update device server 136. The update installation process 200 then proceeds to a state 206 where the client device 104 polls the update device server 136 for the server manifest. In one embodiment, the server manifest may be transferred from the update store 133 to the update device server 136, or the update device server 136 may retain the sever manifest in an onboard memory or storage component for ease of reference. In one aspect, the polled server manifest comprises information used to determine the latest available version of the software, file system, or hardware to be updated. Additionally the server manifest may contain information that describes the size of the update package and other variables used to determine whether an available update is different from the existing file, software component, or firmware present in the client device 104. The polled server manifest may further include an update signature which identifies characteristics of the new code version).

Regarding claim 26

*O'Neill teaches*

continued-execution management part sets first identification information corresponding to second identification information in execution control program beforehand (column 17,lines 45-64, after building the hash array 330 from the existing code version in the state 306, the update creation process 300 proceeds to a state 308 where a new file is opened that will be used to store the instructions used to transform the first code version into the second code version. The new file eventually become the update package 110 which is made available to the servers for

transfer to the clients when updating is requested or desired. The update creation process 300 then advances to a state 310 where the update package 110 is generated. As will subsequently be described in greater detail, the instruction set is formed using a plurality of sequence identification and transformation functions that identify operations, instructions, parameters, or digital information sequences which, when executed in an appropriate manner, will transform the existing code version into the updated code version in an efficient manner. During this state 310, the informational composition and sequence of the existing code version is assessed and instructions are identified to transform the existing code version into the new code version).

Regarding claims 27-29

*O'Neill teaches*

continued-execution management part sets specified time in execution control program beforehand, and causes continuation confirmation communications to be performed from execution control program to continued-execution management part when this preset specified time arrives (column 14,lines 52-67, In one aspect, if the update component 102 is scheduled to operate automatically, then the update component 102 may remain active in the background until the next scheduled operation or alternatively be activated at the time that the update installation process 200 is desirably initiated. The activities comprising the update installation process 200 may further be visible or transparent to the user, according to the developer's preference. Additionally, following the process of polling the server manifest in the state 206 and identifying a difference between the existing version of the file and the latest version available on the update device server 132 in the state 208, the client devices 104 may notify the

user that an update is available and wait for permission to retrieve the update or alternatively, the client device may be configured to automatically retrieve the desired update package (either with or without notifying the user).

***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Anil Khatri whose telephone number is 571-272-3725. The examiner can normally be reached on M-F 8:30-5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Zhen can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

***

ANIL KHATRI
PRIMARY EXAMINER